FIG. 1A

Spreadsheet application 22

Editing cube from spreadsheet 40

Build	Capture

70 45

Intrinsic information analysis tools 225

Cube storage 62

Cube 60

Data interface services 64

FIG. 1B

63

Computing platform

Spreadsheet application 22

Operating system 631

Input device 632

Processor and motherboard 633

Main memory 634

Bus 635

Output device 636

Storage device 637

Network connection 638

FIG. 1C

22

Spreadsheet application

24

Wizard process

40

Editing cube data from spreadsheet

FIG. 1D

22

Spreadsheet application

224

Spreadsheet add-in facility

24

Wizard process

User interface API   226

Programming API   228

FIG. 2

Build 70

- Select cube 82
- Select orientation 83
- 84 For a dimension, select members
- 85 More dimensions? — Yes → (back to select members) / No ↓
- Configure 86
  - Connection 362
  - Spreadsheet 364
- (optional) → Write configuration file 87

FIG. 3A

40

Capture

Spreadsheet selection [41] → Capture multi-dimensional edits [43]

Undo any edits? [44]

— No → Commit [46]

— Yes → Rollback [47] → Commit

Continue? [48]

— Yes → Capture multi-dimensional edits

— No → Done [49]

FIG. 3B

41

Spreadsheet selection

Spreadsheet identification and activation [411] →

Data connections [415]

(cube 60) via ADO [412]

Star Schema via DAO [413]

(cube 60) via ProClarity [414]

FIG. 4

364

Spreadsheet configuration 50

Spreadsheet 52

Data entry sheet

521

Slice controls

522

Column controls

523

Row con-trols

525

Cell range 529

Non-data cell

523

Data cell

(Correspondence 53)

58 **Link sheet**

585

Link cell range

Link cell

589

(Correspondence 55)

56 **Coordinate sheet**

565

Coordinate cell range

569

Coordinate cell

54 **Detail sheet**

542

Data re-connection

544

Data entity

(Correspondence 57)

(Correspondence 59)

65

Cube cell

FIG. 5

43

Capture multi-dimensional edits

431 — User edits a value in a cell associated with a multi-dimensional data entity

42 — Map spreadsheet cell to storage

432 — Parent cell?

Yes→ **433 Parent response**
- 434 Write to deFact table via DAO
- 435 Store information in list

No↓

436 Conventional cell response
- 437 Write to cube via ADO
- 438 Write to deFact table via DAO
- 439 Refresh display

430 — Continue editing?

Yes

No↓

429 — Done

FIG. 6

42

Map to storage

421 — Use the (data cell 529) and the (correspondence 53) to find (link cell 589)

422 — Use the (link cell 589) and the (correspondence 55) to find (coordinate cell 569)

Use the (coordinate cell 569) and the (correspondence 57) to find a corresponding (cube cell 65) in (cube 60)
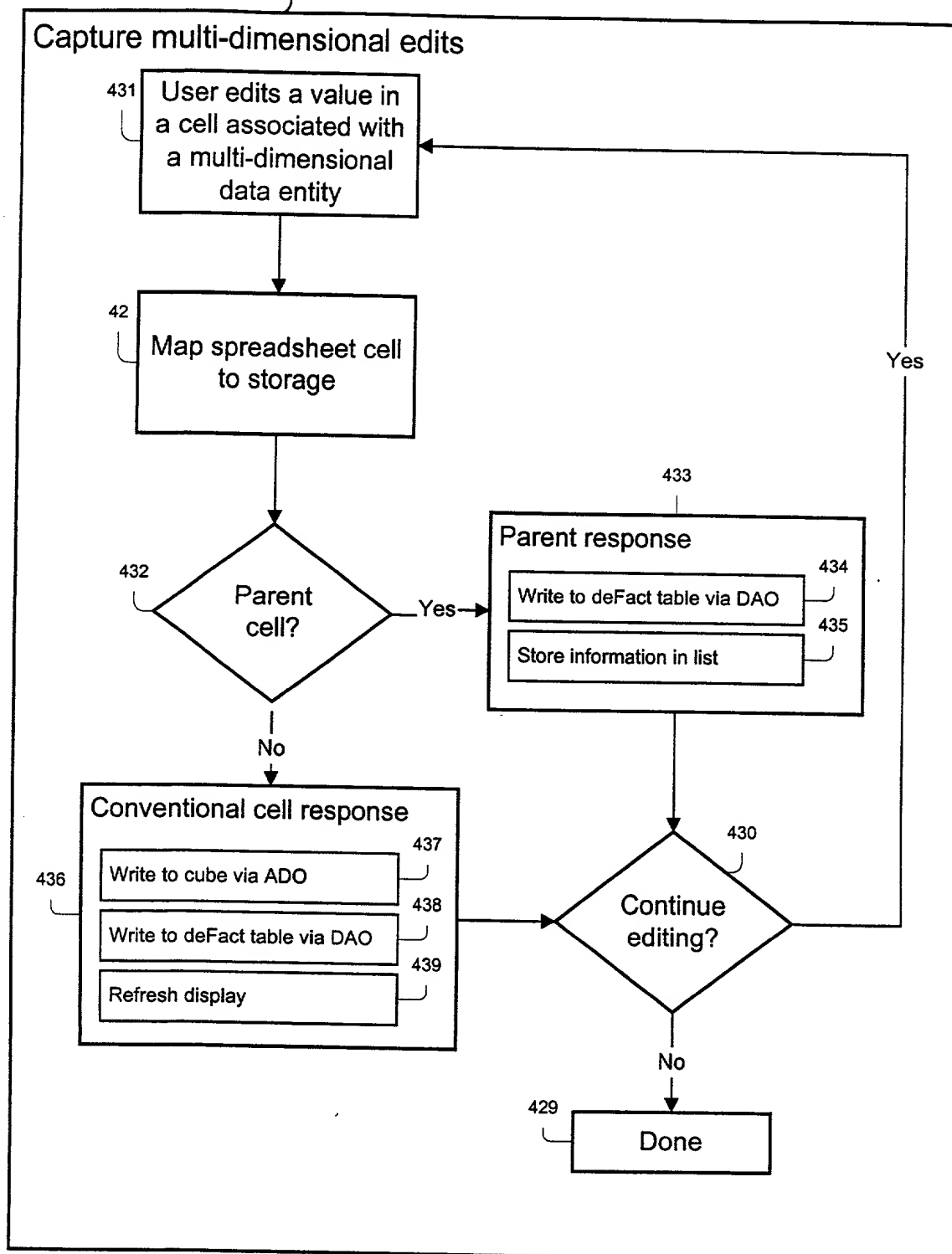
423

569

Coordinate cell

(Correspondence 57)

60 — Cube

cell

65

FIG. 7   Data structures for storage

72

Star Schema

Fact table (71)

Code (711)
Dimension column (712a)
Dimension column (712b)...
Dimension column (712x)
collection of Measure columns... (714)

Hierarchical dimension tables 74x

Hierarachical Dimension Table 74a

Code (741)
Name (742)
Parent (743)
collection of Property columns... (744)

Hierarachical Dimension Table 74b...

Code (741)
Name (742)
Parent (743)
collection of Property columns... (744)

Linear dimension tables 73x

Linear Dimension Table 73a

Code (731)
Name (732)
collection of Property columns... (734)

Linear Dimension Table 73b...

Code (731)
Name (732)
collection of Property columns... (734)
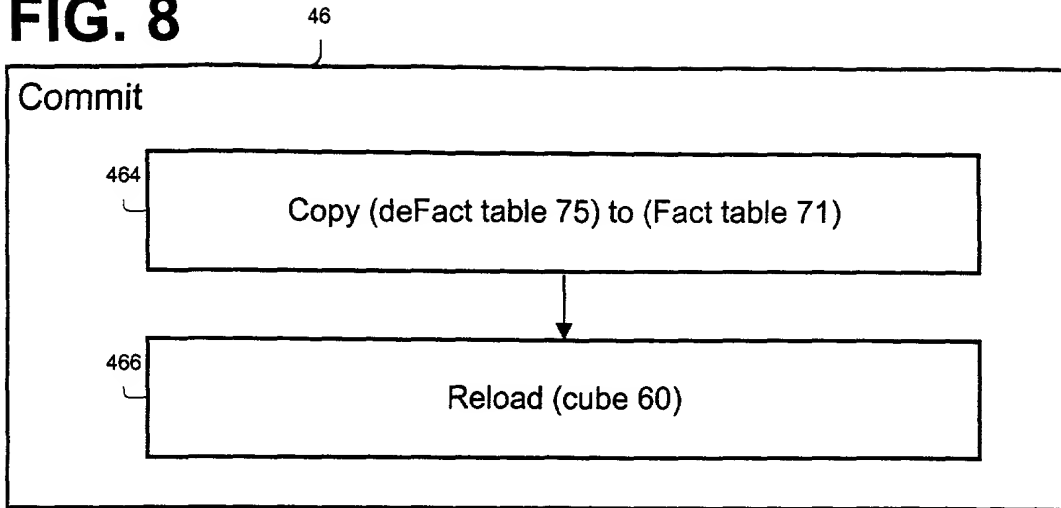
deFact table (75)

collection of Dimension and Measure
columns... (754)
Original value (752)
New value (753)
Comment (756)

60

Cube

cell

65

Dimension 61c

Dimension 61a

Dimension 61b

FIG. 8

46

Commit

464 Copy (deFact table 75) to (Fact table 71)

466 Reload (cube 60)

FIG. 9

47

Rollback

For each change stored as an entry in (deFact 75)

474

476 Write (original value 752) back to (cube 60)

477 Remove entry from (deFact 75)

479 Refresh (cube 60)